

Test driven non-functionals

Wilco Koorn

Stichting Spider, 17 February 2011



1



- My inspiration, my goal
- State of the test driven Union
- What about non-functionals?
- Some examples
- Impact



Copyrighted Material

TOM GILB

PRINCIPLES OF SOFTWARE ENGINEERING MANAGEMENT

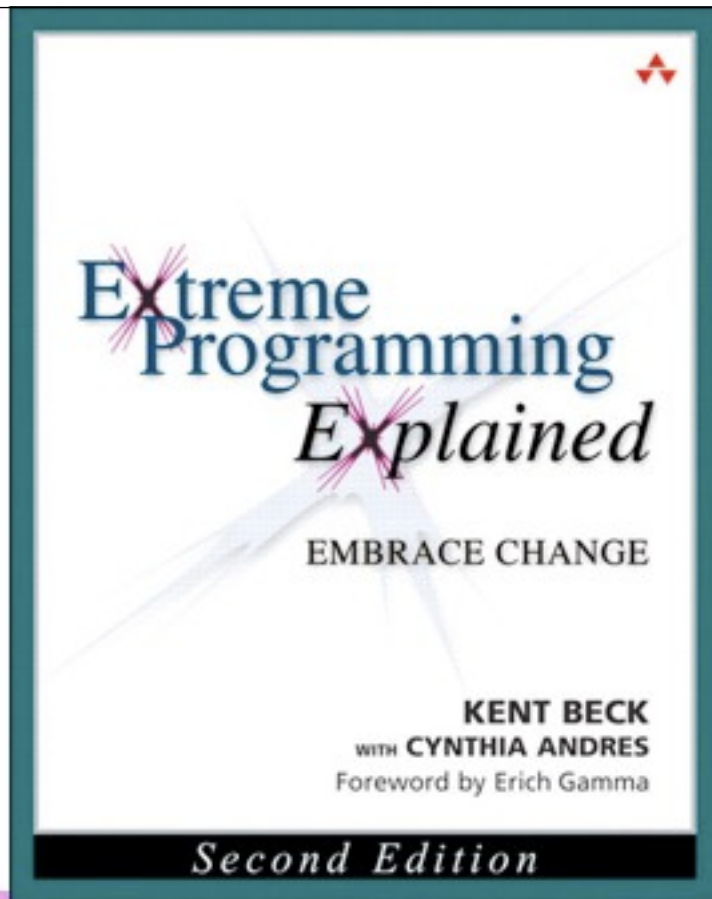


Copyrighted Material



Measure!

- scale
- test
- plan
- now



Test First!
Pair program

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools

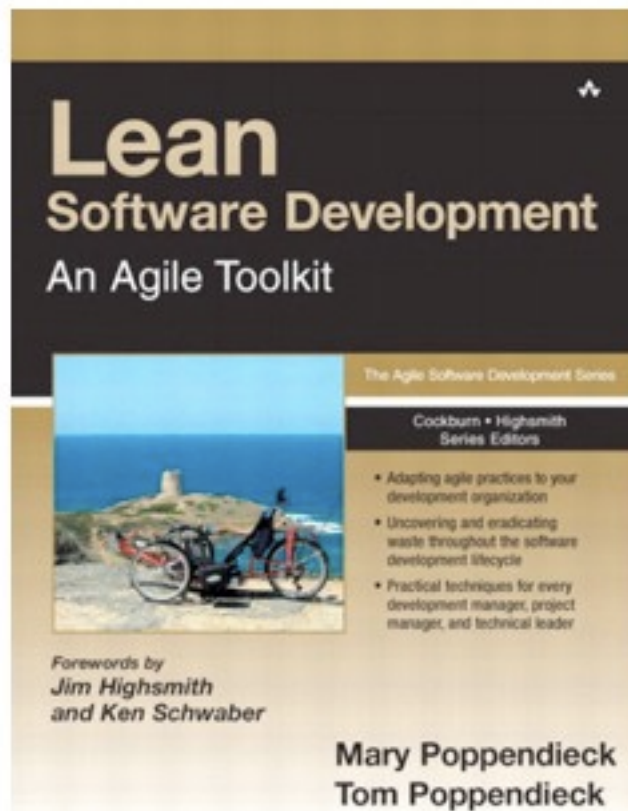
Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.





- Lean!**
- **Eliminate waste**
 - **Build integrity in**

Test driven non-functionals



13

Why?

How?

14



15

Test Driven Development

- First, build a test
 - Then code **"The simplest thing that works"**
- Build a new test that breaks the code
- Improve code so it works again
- ...
- Until **the pair** can't think of more tests that will break the code

16

The simplest thing driving 20 km/h?



17



Refactor

18



19

Test and Code go hand-in-hand



20

“Test driven development” is Common Practice in Java world

- Pair programming
- Test a little, code a little
- JUnit, Fitnesse, ...
- Tests also used in regression testing
- Build never breaks
- Tend to do too much

Non-functionals

- Performance
- Manageability
- Recoverability
- Security
- Scalability
- Usability
- ...

“Test driven development” for non-functionals is Uncommon Practice

- Testing can be automated e.g. for:
 - Performance
 - Maintainability
- Testing hard to automated for:
 - Usability
 - Documentation

23

How?

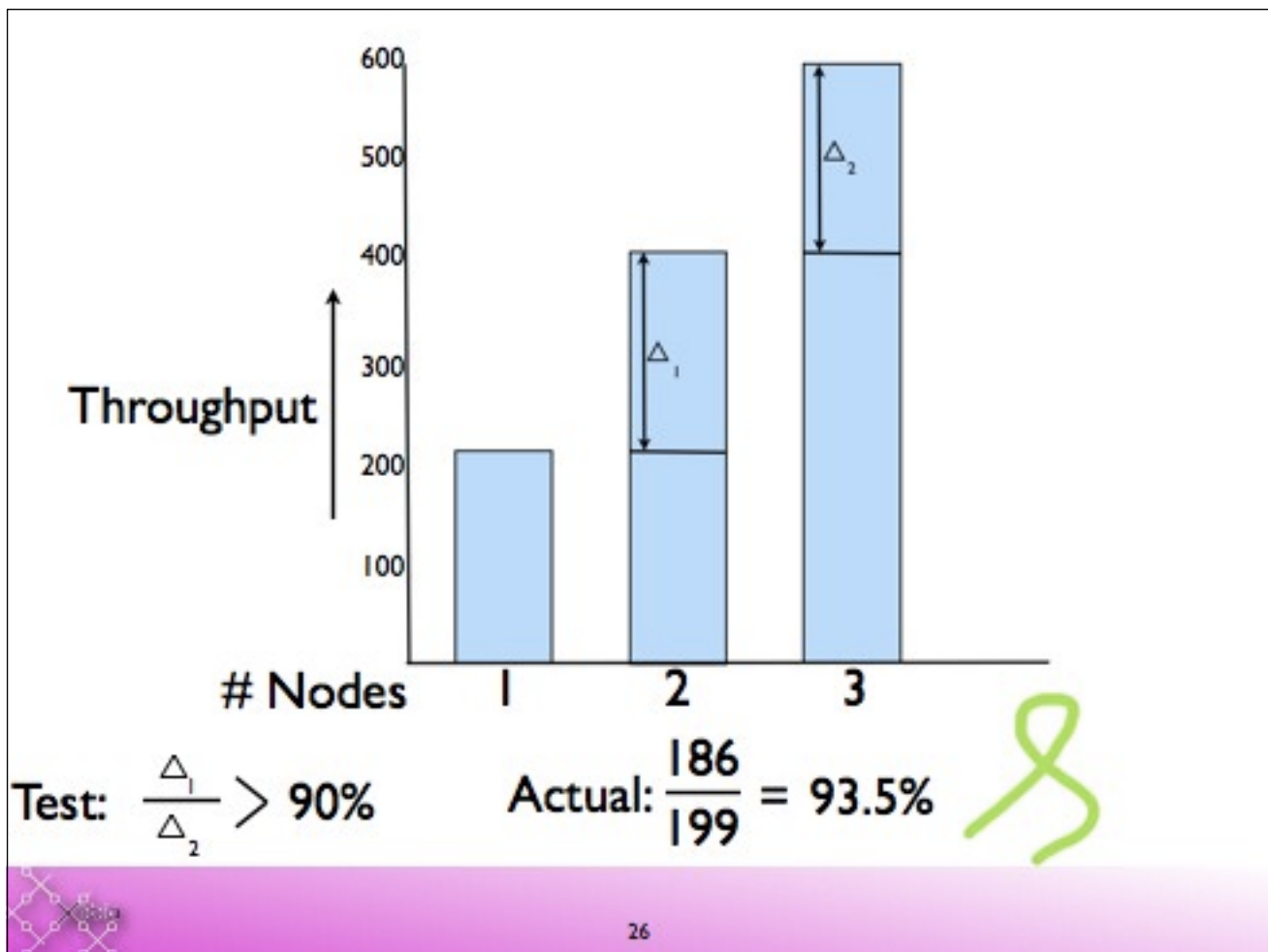
- Keep regression aspect
 - Run nightly, or on separate systems
- Make test binary, or use a range
- Use trend analysis
- Interpret trend over a period of time

24

Binary test example: Scalability

- All components have HTTP interfaces
- Build a test that measures throughput
- Deploy on single node, measure
- Deploy on two nodes, measure
- On three, measure
- Ultimate test: do results form a “line”?

25



26

Binary test example: Quality

- All components are JUnit tested
- Code coverage is measured
- Coverage trend is watched, over a period of time
- Binary test: does the trend go up?

27

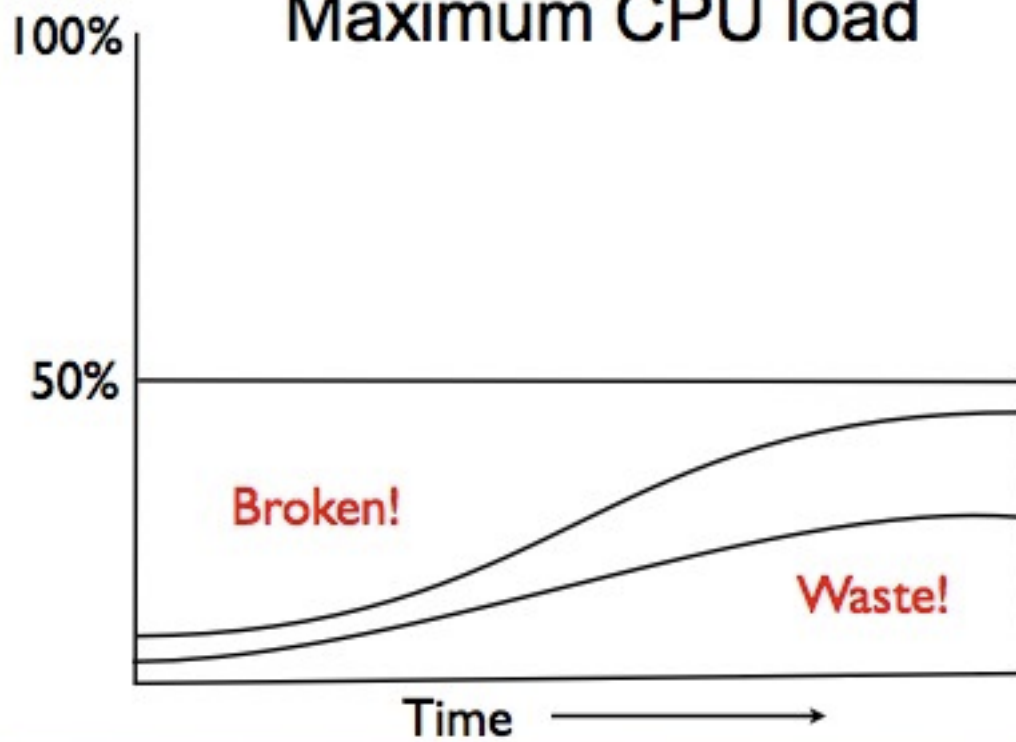
Binary test example: Component Quality

- Cobertura reports 75.1% code coverage
- Cobertura reported 69.7%% last week
- Trend over a week goes up



28

Range test example: Maximum CPU load



29



30



31

Organizational impact



32

Ops and Dev go hand-in-hand



33



Summary

Test driven non-functionals is uncommon

Applicable when automatable

Part of regression test suite

Attend to preventing waste

Organizational impact



Wilco Koorn

wkoorn@xebia.com

06-10376212

